

Inventory Optimization & SAS

Palladian Analysis & Consulting LLC

Houston, TX

Tom Taylor
November, 1997

Inventory, Boring Hah! Easy Hah!

In our perceived world of just-in-time inventory it is surprising how little just-in-time inventory there really is. Only high volume processes lend themselves to just-in-time inventory. Most of the world is surprisingly low volume or so discontinuous in nature as to prohibit the easier part of just-in-time processes. This paper speaks to the world that is not the highly publicized high volume problems.

The General Types of Inventory Systems

The most common types of inventory are: continuous flow; high quantity, low variety; and high variety, low quantity. Continuous flow speaks to systems where volumes are so high that fractional quantities can be ignored (e.g. Auto plants) or that the material can be handled in fractional quantities (e.g. oil refining). These classes of problems are relatively easy and have by and large been solved with some degree of success. High quantity, low variety, is typically a manufacturing or distribution operation. When the system has few new products and/or few new customers, these are easy problems. In the more common growing dynamic situation, keeping up with change management can make these problems very complex. High variety, low quantity, are the consistently difficult problems. Fraction rounding becomes a major issue in these problems. Growing, dynamic changes in customer lists or product lists add to the complexity. This paper focuses on the non-continuous flow inventory problems.

Two additional inventory systems 'low quantity, low variety' and 'high quantity, high variety' will be mentioned but not discussed in any detail.

Definitions of Quantity and Variety

For this paper quantity means units per time period. A time period could be a day, week, or month. When high quantity is referenced, this means at least 30 units per time period. It could also be 30,000 units per time period. 30 units a month may not sound like a lot but it is still a large enough quantity that statistics work in favor of the problem solver and good answers can be obtained relatively easily. If 30 units a month is restated as one unit per day then the problem becomes very complex because statistical techniques stop working at this volume. Stating the proper time period for the problem is critical.

Variety means different active inventory items. Low variety would imply less than several hundred different active units in a time period.

Classifications of Inventory Types

Item Variety	High	Airplane Parts Auto Parts Clothing Retailers	Grocery Stores Dry Goods Retailers
	Low	Job Shops	Auto Assembly Plant Oil Refiner Chicken Processor
		Low	High
		Item Quantity	

High Quantity, Low Variety, Inventory Problems

These are typically the type of problem a manufacturer would face. A manufacturer here could be a component manufacturer for the automotive or computer industry. Or it could also be a supplier of specific items to retail e.g. suits or candles. The quantity of each individual item is relatively high, more than 30 per time period. Statistical techniques start to be very useful in this class of problems.

Demand shocks are the major problem area with these systems. In a manufacturing situation seasonality, new product opportunities, or new customer opportunities can cause demands shocks. Without demand shocks, forecasting is a simple problem for products like SAS ETS. The demand shocks are typically the bulk of the problem.

There are many older and simple systems, which have simple fixed restocking levels and times for restocking. The world is seldom static enough for these systems to work without extensive system maintenance.

High Variety, Low Quantity, Inventory Problems

This type of problem has two common venues. Individual retail store level inventories

are the most common type of problems. The second type are referred to as high cost, low usage industrial items, airplane parts being a good example.

For stocking individual retail stores, data systems today are rapidly improving. The improvement is largely due to the availability of cheap microprocessors and bar coding. Because sales patterns vary from store to store and units stocked are usually below 30 units per item, good inventory management at the store level is still one of the more widely unsolved complex problems. Most current systems are typically solved by a 'standard' inventory model for all locations. Dealing with local market nuances becomes problematic. The standard model usually errs on the side of sending out too much initially and later reallocating inventory among the retail locations. This is potentially an extremely labor intensive process. This process probably costs 1% of an item's retail value each time it is reallocated. Thus the tendency is to be heavy in inventory and dispose of excess through local sales.

The high cost, low usage, parts are generally managed by having a database of global stock quantities. The individual part may be stocked as ten in one warehouse or one each in ten warehouses, depending on distribution needs. This system can be easily managed except when new products come on line and older parts are going obsolete. The new product case is usually handled by requiring the new product developer to recommend stocking levels. The obsolesce case is usually mishandled because awareness of obsolesce is seldom handled by standard systems in an organization. A periodic survey of obsolesce plans would take care of this situation for most organizations.

High Variety, High Quantity, Inventory Problems

The example that should come to mind here is the common grocery store. Big stores stock 20,000 to 50,000 different items and commonly turn inventory in excess of 30 times a year. Volume sales and rapid reorder tends to mask inventory problems to the consumer. Department managers and stock boys often scramble to cover the miscalculations. One source of the miscalculations are the old centralized shelf space planning systems. These systems do not easily adapt to each store's unique situation. As a result stock levels are often kept 20-30% too high and mask the problem. With cheap powerful computers, it is now economical to produce dynamic shelf allocation plans at the store level.

Low Variety, Low Quantity, Inventory Problems

These are typically job shops, e.g. machine shops, fabrication shops. The inventory numbers are small enough that just about any spreadsheet will handle inventory for these businesses. The complex issue in these businesses is usually scheduling of work centers and personnel. These scheduling problems are often more difficult to solve than inventory problems, but scheduling is not the subject of this paper.

Keys to Successful Development of Inventory Optimization

Inventory problems tend to be company specific. One size-fits-all software packages e.g. retail solution, seldom handle inventory problems well. (100 years ago, retailers stocked one shirt size and arm garters for adjusting the size. This may be a metaphor for retail inventory solution packages.) Non-inventory business systems such as ordering systems and accounting systems do lend themselves to standard solutions because of GAAP (Generally Accepted Accounting Practices) and tax standards. On the other hand, inventory systems can really be a strategic advantage. Thus having customer specific inventory solutions written in a general purpose package such as SAS has many advantages.

Allowing for the unexpected is the rule when dealing with inventory. Inventory sales, restocks, transfers, disappears, reappears (admittedly disappearing more than reappearing) the possibilities are endless. Good inventory systems allow for all cases and needs to follow the famous 80-20 rule. It should automatically handle 80%+ of the cases (which is probably 99%+ of the volume). For the remaining cases, a human editing the inventory levels directly is typically the most efficient solution.

Adapt the system customer mindset. Customer organizations tend to rely on the traditional way of doing things, if the new systems does not accommodate the traditional way in some fashion, the customer will not accept the new system. If new methodology is being introduced, the system should be able to also replicate the old system so that comparisons may be made between results.

SAS Tools Most Useful for Inventory Optimization

ETS (Estimating Time Series) for forecasting is flexible, and allows several models to be tried quickly. It is quick, easy to use, and is always a good starting place when any type of forecast is involved.

OR (Operations Research) for optimizing is a good, solid linear programming & integer programming tool. Many inventory problems turn into integer programming problems, which are relatively easy to formulate, but can be impractical to solve due to their heuristic search solving methodology.

SAS-Connect is a product which, among other things allows remote execution of SAS jobs on remote servers. For most inventory problems it is better to keep large data sets and heavy number crunching on the remote machine.

Two Example Applications of Inventory Optimization

Apparel Application (High Variety, Low Quantity)

The objective is to optimally distribute affinity merchandise among 100's of stores. Affinity merchandise is merchandise that if 2 of size A sells then 4 of size B should also sell. Stores tend to stock 1 or 2 of each item, there can be 50 sized-items in an affinity group. To make things more difficult, each store can have a unique affinity profile! It turns out that markets actually need different size groups in different parts of the same city. In this example, there is a possible 8000 affinity groups. While total inventory per store is generally less than 2000 units, a store is typically stocking only about 200 affinity groups at any time. Clearly, most affinity groups are not stocked in all fifty sized-items.

The apparel problem is each time an affinity group was distributed it required about 20,000 individual product decisions to correctly distribute 3000 units of merchandise. The decision process needed to be completed in 3 hours.

The traditional solution was to use lots of standard rules-of-thumb e.g. send 20 units to every store, period. The problem with this solution is it caused lots of unsold merchandise. Managers had to spend time at some later date sorting out the excess merchandise or moving it to a store where it would sell. One merchandise distributor tried to do a detail solution by using a spreadsheet and looking at the individual store affinity groupings. He quit after 8 hours with the problem about 30% complete.

The SAS solution first calculates the affinity scale for each store. Then using SAS-OR (Linear Programming) merchandise is optimally distributed while considering factors of existing inventory, sales volume and affinity scale. To make the software easy to use for the non-programmer, SAS-AF generates flexible set-up tables for all factors. SAS-Connect is used to allow for a client-server setup to allow keeping of large data sets and OR processing on the server. Key results are exported to a spreadsheet to allow user to edit any results so that one time special situations can be incorporated into any result. SAS gives a detailed solution in about 1 hour, depending on the number of stores involved. Problem setup takes about 30 minutes.

Results, the problem of mis-allocation of stock have disappeared as inventory cycles through the system. Labor, which was previously consumed reallocating inventory, has been eliminated.

Manufacturing Application (High Quantities, Low Variety)

In this situation the manufacturer has several individual but related product lines e.g. milk and butter. Frequent production forecasts are needed but there is no practical way to completely automate the forecasting system. Forecasts are based on some historical factors and important new trends and client information do not fit well into classic forecasting systems

The traditional solution was to make quick spreadsheet estimates by using many simplifying assumptions. These solutions were difficult to defend when they later proved wrong and the scapegoat hunt was on.

The SAS solution was to establish a system of using ETS to provide an initial forecast and provide for easy adjusting based on non-historical factors such as one time orders, new customers, promotional pricing or new products. The SAS solution provided many improvements over spreadsheet solution. Access to data became automatic. This was a time saver by itself. The solution was for the first time multi-user as people made forecasts it was updated in the central database. A system of documenting who made the forecasts was considered but discarded as too bureaucratic. Non-forecastable adjustments were categorized and a quick systematic coding was arranged for the typical situations. Ultimately the whole system was explained to the entire sales & marketing team so that forecast meetings could be focused on the key forecast factors not drift into a discussion of personalities. Finally by removing the spreadsheet from the process, the little errors/changes which always drift into spreadsheet formulas and are forgotten were completely eliminated.

Results, now the forecasts can be replicated and discussions of “how did you come up with that number” have stopped. There is less slack in forecast and occasional one-time refinements are made and tested in a systematic way. Factory overtime is down and obsolete inventories problems have declined.

Time-to-Develop Inventory Optimization Solutions

The largest time eater is often user testing! It is not software development. The testing phase can be rather expensive, as chasing down problems in local data is very time consuming. Local data is often not current. Official accounting often has adjustments, which has not been reincorporated in local data.

Screen development via rapid prototyping normally takes two or three passes. Traditional development systems try to carefully define the problem and have users sign off on all details of problem definition and screen presentation. In reality users reject 90% of initial attempts to solve their problem. Shoving a signed agreement under their nose and telling them they got what they asked for seldom helps the situation. The fundamental problem is that people learn over time! Needs change, thinking evolves, markets change. User needs are always a moving target. It is often better and faster to allow for iterations in the development of major projects. Two or three passes at screen development will probably happen whether planned for or not, so plan for it! Rapid prototyping of screens is easy with SAS-AF. Users will be more satisfied. The project will be more successful.

Conclusion

Inventory problems are seldom simple problems, but improved management of inventories is paramount to the profitability of most businesses. Working closely with the end users to develop a system which optimizes the strengths of the specific enterprise is the key to maximizing system value. Given the readers general knowledge of the SAS system and the brief comments on SAS-ETS and SAS-OR presented here, the writer has attempted to expose the excellent tools for optimizing inventories that are contained in SAS. An application specific, inventory optimization system can become a formidable competitive advantage for many enterprises.